[2345/173]

SYSTEM AND METHOD FOR TESTING THE LOAD OF AT LEAST ONE IP-SUPPORTED DEVICE

The present invention is directed to a system, as well as to a method for testing at least one device in a communications network that is based on an IP (Internet protocol) standard, in the loaded state.

Developers, system providers and administrators of large heterogeneous network configurations based on the IP standard, such as the Internet, are facing mounting challenges due to the rapid growth of the Internet and the speedy advancement in transmission and hardware technology. Therefore, testing a new network or network expansions prior to actual installation, for their load state, is considered an important task. By a load test, one understands, quite generally, the targeted loading of the network, particularly of the routers and servers connected to such a network, in order to determine their performance with respect to the required data throughput and the response time to a user request. There is, therefore, a need for a test system, which is able to test network components based on the IP standard, under real load conditions, to be able to ensure that all network components are functioning properly in an error-free manner, within their predefined performance limits.

The object of the present invention is, therefore, to devise a test system and a test method, which will be able to be adapted simply and quickly to changing IP network structures, IP access techniques and IP communications protocols, and run by a single operator.

The present invention achieves this technical objective, first of all, by employing the features of Claim 1.

A core idea of the present invention is to provide a semi-automated test system, which is able to establish a plurality of mutually independent IP connections to a communications network based on the IP standard, in order to run, via these connections, mutually independent test procedures, which each correspond to the operations of a real network user. In very general terms, such a test system is designed for testing at least one device, in the loaded state, in a communications network based on the IP standard.

To this end, the test system includes at least one programmable control device having an assigned memory device in which a plurality of session scripts may be stored, each of which contains a predefined test procedure. In all the documents, a session script is understood to mean the scripted simulation, in recorded form, of a real network user, who typically performs actions based on the IP standard, such as establishing a connection to a provider, downloading files to a server, using a web browser, and initiating the connection. A session script may contain, for example, a user ID, a user password, an IP destination address, for example of a server connected to the communications network, the user ID and the password of such a server and the service and communications protocol utilized, such as the FTP (file transfer protocol) or the HTTP (hypertext transfer protocol). It is important to point out that each session script contains a predefined number of operations that a real user could enter into a personal computer in order to request a specific IP service via the communications network.

In addition, at least one session computer is connected to the control device. Each session computer has a plurality of mutually independent connection interfaces, via which an independent IP connection to the communications network may be established at any one time. Assigned to each connection interface, in turn, is a script-processing device, also called load-generating device in the following, which, in dependence

upon a session script assigned by the control device, may establish an IP connection to a device to be tested and start the predefined test procedure. This makes it possible to run a plurality of mutually independent test sessions in automated fashion between various simulated users and one or more devices connected to the communications network, such as of a router or a server, without an operator having to manually carry out a session at the session computer.

The complexity of the test system may be enhanced by applying the same or different session scripts to a plurality of load-generating devices of a session computer, which, in dependence upon the session script assigned in each instance by the control device, are then able to establish a separate IP connection to one or a plurality of the devices to be tested and initiate the corresponding test procedure. To this end, implemented in each session computer is a session-management device which supplies the session script assigned by the control device to each load-generating device.

The session computers are designed to support every existing network-access technology. They will be able to be readily adapted to future network-access technologies. For example, every connection interface of a session computer may be connected to an analog and/or digital modem. It is also practical to insert one or more interface cards, for example LAN cards, into the session computers, which each have a plurality of connection interfaces. On the other hand, each connection interface of a session computer may be assigned to an analog or digital modem or be linked to a conventional concentrator to interface to an ATM (asynchronous transfer mode) network. As digital modems, ISDN modems or ADSL (asymmetric digital subscriber line) modems come under consideration. In this manner, a separate IP connection may be established via each connection interface of a session computer.

The control device and the session computer connected thereto may either be implemented in one single machine or be connected via a backbone network.

To be able to log and later analyze the various test sequences, each session computer has a memory for storing status data on each device to be tested and results and status messages from each initiated test procedure. The status data of a device to be tested are considered, in particular, to be the data throughput from and to the device to be loaded, as well as its response time. The response time of a device is understood, in this context, to be the time that the device requires to react to a specific request from a user.

The session computers transfer the stored status data on the tested devices and the results and status messages from each active test procedure to the control device which is able to display this data on a display device assigned thereto and analyze the same. In addition, the control device has a keyboard assigned to it, via which one may enter new session scripts, for example, or intervene in active test procedures in order, for example, to abnormally terminate a test procedure or reset parameters. In this manner, the test system may be adapted to any hardware and software change in the communications network, merely by writing a new session script and storing it in the control device.

The communications network based on an IP standard is, for example, the Internet or any firm-specific Intranet. As devices to be tested, access routers and servers come into consideration, for example, which belong to various service providers. Servers, which are based on an IP standard, are generally known and are, therefore, not discussed in detail.

The technical objective is likewise achieved by the method steps of Claim 10.

An advantageous further embodiment constitutes the subject matter of dependent Claim 11.

The present invention is elucidated in the following on the basis of an exemplary embodiment, in conjunction with the enclosed figure.

The figure shows a test system, denoted by 10, with whose assistance, one may test the operability, for example, of the Internet 90, in particular of its network components, such as access routers 80, or of servers 100 of various service providers connected thereto, in the loaded state. Test system 10 may also be referred to as an IP load-test system, to indicate that the test system, as well as the components to be tested with respect to their load, support IP protocols. Test system 10 includes a control and service computer 20, to which a plurality of test computers, called "session computers" in the following, are connected, in the present example, via a star coupler 30 and a so-called backbone network 35. For the sake of clarity, merely three session computers 40, 50 and 60 are schematically depicted, session computer 40 being shown in greater detail. For that reason, the design of the session computers in terms of circuit technology is principally explained with respect to session computer 40.

Control and service computer 20 has a keyboard, by way of which an operator may generate any permissible session scripts, for example, which are subsequently stored in a memory (not shown) assigned to control and service computer 20.

By session script, one understands, quite generally, the description of an automated user, who, in conjunction with a session computer, may automatically execute IP-supported operations and activities, which a real Internet user could also undertake manually at a personal computer. In other words, each session script contains a defined test procedure,

which may be used to simulate a typical behavior of a real Internet user at the session computers. Each session script includes an initialization procedure, a test procedure, as well as an end procedure. In this context, the initialization and end procedures are executed only once in each session, while the test procedure may be carried out repeatedly. In addition, the writer of the session script may include certain error and status messages which are generated over the course of a running test procedure. In principle, any operations and actions at all may be utilized in a session script, as long as they are based on the IP standard. Moreover, by way of control and service computer 20, variables may also be set within a session script before the test procedure is started. In this manner, generally formulated session scripts may be easily and quickly adapted to special customer requests. Other parameters, such as the number of repetitions of one test procedure and timing intervals, may likewise be defined in a session script.

Moreover, an operator at the control and service computer 20 may determine at which session computer(s) and via which connection interfaces of the selected session computers, a test procedure should be started, how many test procedures should be started at the same time, how long a test procedure lasts, or how often the same test procedure should be repeated.

Inserted into each session computer 40, 50 and 60 are, for example, four LAN cards 42, 52 and 62, which, in turn, each have four separate connection interfaces 441-44n, also called connection ports. To interface with Internet 90, in the present example, each connection interface is connected to a digital ADSL modem 70, although such an interface connection is only shown for connection interface 44_1. Each modem 70 may be linked via a transmission line to an access router 80 or to various routers. It goes without saying that test system 10 may also support any other access technology. Thus, instead of

ADSL modems, ISDN routers may also be connected to the connection interfaces of the session computers. It is also conceivable to link connection interfaces of any one session computer to a generally known concentrator, which provides an access to an ATM network. In another case, it is possible to link the connection interfaces via a serial connection to analog or digital modems, with whose assistance a dial-up connection to any router and, thus, to Internet 90 may be established. As described with reference to session computer 40, a script-processing device, in the following also named load-generating device 45_1-45_n, is assigned to each connection interface 44_1-44_n of each session computer 40, 50, and 60, as explained in greater detail further on. It should be noted here that the load-generating devices may also be implemented as software modules.

In addition, in each session computer 40, 50 and 60, a session-management device is provided, whose task is to supply session scripts assigned by control and service computer 20 to selected load-generating devices. With regard to session computer 40, session-management device is denoted by 46. In addition, in each session computer 40, 50, and 60, a memory may be provided in which the status data from the devices to be tested, as well as the results and status and error messages from the initiated test procedures are stored. These status data, status and error messages, and results of the test procedures in question may be transferred from each session computer to the control and service computer 20, and stored there. In addition, the control and service computer is designed to analyze the messages and results received from the session computers and to graphically display the same via a monitor.

At this point, it should be noted that test system 10 may be used to test IP networks with respect to their software and hardware components from various manufacturers. It is, thus, possible to test the operability of routers and servers within

a network based on an IP standard, using one single, central test system.

A scenario is used in the following to elucidate the method of functioning of test system 10.

The intention here is for test system 10 to check the handling capacity of server 100 connected to Internet 90. Here, the assumption is initially made that router 80 is functioning in an error-free manner, so that error messages occurring during the test procedure may be clearly attributed to server 100 to be tested.

The assumption is also made, based on the manufacturer's specifications, that server 100 is able to handle up to 50 users simultaneously, who, for example, want to download data via the FTP protocol. In this case, the operator at control and service computer 20 selects that session script which makes it possible to automatically establish a connection to server 100 to be tested, and download a dedicated file from a dedicated directory of the server. If, in the present test case, 32 queries to server 100 are simultaneously simulated, at control and service computer 32, the operator selects connection interfaces via which a test procedure should run in each instance. For this, the addresses of the connection interfaces are either entered via the keyboard of control and service computer 20, or appropriate icons are clicked on at the monitor. For example, the operator selects all 16 connection interfaces 44_1-44_n of session computer 40 and the first eight connection interfaces of the two other session computers 50 and 60, respectively, via which a test procedure should run to server 100. Control and service computer 20 subsequently transfers the session script in question and the addresses of the selected connection interfaces to the particular intended session computers. At this point, the session-management device in each session computer assures that the session script is loaded into all load-generating

devices 451-45n of session computer 40, as well as into the first eight load-generating devices of session computers 50 and 60, respectively. Under the control of the session script, each load-generating device establishes an IP connection via the connection interface assigned to it and ADSL modem 70 connected thereto, for example via the PPPoE protocol to router 80, which assigns each connection interface its own IP address and a user password. After that, an identification is carried out between server 100 and the particular intended connection interface via protocol PPP. Following this initialization phase, each selected load-generating device is prompted by the session script to execute the FTP IP service, which prompts the server to download the files in question to the selected connection interfaces. The test procedure is subsequently terminated by each selected load-generating device, and the connection is released. During the individual test procedures, predetermined status and error messages are logged in the session computers to the selected connection interface messages and, at the same time, routed to control and service computer 20, to be able to monitor the running test procedures there. Each session computer 40, 50 and 60 is able to determine the data throughput, as well as the response time of server 100. Since the average data throughput from and to server 100, as well as the server's response time are preset by the manufacturer, it may be determined from the calculated data throughput and from the ascertained response time, for each selected connection interface, whether server 100 has executed the 32 test procedures with or without errors. In this manner, each Internet component may be automatically tested by test system 10 with respect to its required performance features, in that appropriate session scripts are loaded into selected load-generating devices of the particular intended session computers, and are executed.

Since the connection interfaces and the load-generating devices of each session computer assigned thereto are designed independently of one another, in the present example, users,

who become active independently of one another, may be simulated using each session computer 16. To all intents and purposes, a single operator at control and service computer 20 suffices in order to be able to operate a test system having any number of automated users.

Thanks to test system 10, it is possible to automatically test a device to be tested to check the load produced by a plurality of network users. To this end, it is merely necessary for an appropriate session script to be written for each permissible user action, and to be stored in control and service computer 20. Any test situations at all may be simulated by loading appropriate session scripts onto selected load-generating devices of the particular intended session computers 40, 50, and 60, which then, independently of one another, establish separate IP connections to the devices to be tested, and execute test procedures thereon.